**SPEECH EMOTION RECOGNITION USING MACHINE LEARNING**

## A MINI PROJECT REPORT

*Submitted by*

|  |  |
| --- | --- |
| **SAKAMURI ANILA CHOWDARY** | **(111520104130)** |
| **SANGANA SWARNA** | **(111520104132)** |
| **SHAIK SAMEEHA TABASSUM** | **(111520104141)** |

***in partial fulfillment for the award of the degree***

***of***

## BACHELOR OF ENGINEERING

***IN***

## COMPUTER SCIENCE ENGINEERING



**R.M.D. ENGINEERING COLLEGE**

**(An Autonomous Institution)**

## KAVARAIPETTAI - 601206

**JUNE – 2021**

# BONAFIDE CERTIFICATE

Certified that this project report titled *“***SPEECH EMOTION RECOGNITION USING MACHINE LEARNING***”*, is a Bonafede work of **SAKAMURI ANILA CHOWDARY (111520104130), SANGANA SWARNA(111520104132)** and  **SHAIK SAMEEHA TABASSUM (111520104141)** who carried out the work under my supervision

## SIGNATURE SIGNATURE

**Dr. P. Ezhumalai, B.E, M.Tech, Ph.D., Dr. Sherine Glory, B.tech., M.e., Ph.D.**

**HEAD OF THE DEPARTMENT**, **ASSISTANT PROFESSOR**,

Department of Computer Science Department of Computer Science Engineering, Engineering,

RMD Engineering College RMD Engineering College

R.S.M Nagar R.S.M Nagar

Kavaraipettai – 601 206. Kavaraipettai – 601 206.

# VIVA-VOCE EXAMINATION

The Viva - Voce Examination of the following students who have submitted this project work held on………………

|  |  |
| --- | --- |
| **SAKAMURI ANILA CHOWDARY** | **(111520104130)** |
| **SANGANA SWARNA** | **(111520104132)** |
| **SHAIK SAMEEHA TABASSUM** | **(111520104141)** |

**INTERNAL EXAMINER EXTERNAL EXAMINER**

**ACKNOWLEDGEMENT**

I take this opportunity to put to record my sincere thanks to all who enlightened my path towards the successful completion of this project. At the very outset, I thank the almighty for his abundant blessing on me.

It is my greatest privilege to convey my thanks to Chairman

**MR. R. S. MUNIRATHINAM** and Vice-Chairman **MR. R. M. KISHORE**

**B.E., MBA.**, for having provided me with all required facilities to complete my project without hurdles.

We take this opportunity to give profound and heartfelt thanks to the Dean- Research **Dr. K.SIVARAM B.E., M. Tech., Ph.D.**, and Dean-Academic **Dr. K. K. THYAGHARAJAN B.E., M.E., Ph.D.**, for their continuous support in successful completion of this project.

It is my greatest privilege to convey my thanks to Principal **Dr. N.ANBUCHEZHIAN M.S., MB.A., M.E., Ph.D.**, for having provided me with all required facilities to complete my project without hurdles.

I express my sincere feeling of gratitude to our Head of Department of Computer Science and Engineering **Dr. P.EZHUMALAI B.E., MTech, Ph.D.**, for his advice and encouragement in all stages of my project work.

It’s my pleasant duty to deliver hearty thanks to the extraordinary efforts of our project guide, **Dr. SHERIN GLORY, B.E, M.Ed., Ph.D.**, Assistant Professor, Computer Science Engineering for his continuous guidance and suggestions without whom this project would not have become a reality.

I owe my sincere thanks to Computer science and Engineering department staff of

R.M.D. Engineering college for their constant support and encouragement towards the completion of this project.

## ABSTRACT

Emotion recognition from speech signals is an important but challenging component of Human-Computer Interaction (HCI). In the literature of speech emotion recognition (SER), many techniques have been utilized to extract emotions from signals, including many well-established speech analysis and classification techniques. Deep Learning techniques have been recently proposed as an alternative to traditional techniques in SER. This paper presents an overview of Deep Learning techniques and discusses some recent literature where these methods are utilized for speech-based emotion recognition. The review covers Environments and databases used, graph representation of particular emotion extracted, contributions made toward speech emotion recognition and limitations related to it.

|  |  |  |
| --- | --- | --- |
| **SNO** | **TABLE OF CONTENTS**  **TITLE** | |
|  |  | |
| 1 | **INTRODUCTION** | |
|  | * 1. Speech emotion   2. Techniques for SER   3. Deep learning techniques of SER | |
| 2 | **LITERATURE SURVEY** | |
|  |  | |
| 3 | **PROPOSED WORK**    3.1 LTSM Deep Learning | |
| 4 | **COMPONENTS** | |
| 5 | **SOFTWARE REQUIREMENTS** | |
|  |
|  |
| 6 | **SOURCE CODE** |
| 7  8    9  10 | **NOVELTY OF THE PROJECT**  **EXPECTED RESULT**  **CONCLUSION AND JUSTIFICATION**  **REFERENCES** |

# INTRODUCTION

## 1.1 SPEECH EMOTION

Emotions plays a main role in one’s life. We can calculate the state of a person by just observing their mood and the emotion that they are carrying at that situation. This helps us to behave nicely and to grab the attraction. One of the way of calculating the emotion is by speech ie.., the way they speak.

Speech emotion is one the way of finding one’s emotion using speech. Emotion recognition from speech has evolved from being a niche to an important component for Human-Computer Interaction. There are many paths like machine learning, neural networks etc.. but now a days deep learning is used to calculate the emotions using speech.

Amongst the numerous models used for categorization of these emotions, a discrete emotional approach is considered as one of the fundamental approaches. It uses various emotions such as anger, boredom, disgust, surprise, fear, joy, happiness, neutral and sadness.

**1.2 TECHNIQUES FOR SER**

Another important model that is used is a three-dimensional continuous space with parameters such as arousal, valence, and potency. The approach for speech emotion recognition (SER) primarily comprises two phases known as feature extraction and features classification phase. Generally we call it as training the model and classifying the data or testing the data. The most commonly used linear classifiers for emotion recognition include Bayesian Networks (BN) or the Maximum Likelihood Principle (MLP) and Support Vector Machine (SVM). There are many non-linear classifiers available for SER, including Gaussian Mixture Model (GMM) and Hidden Markov Model (HMM). These are widely used for classification of information that is derived from basic level features

* 1. **DEEP LEARNING TECHINIQUES OF SER**

Deep Learning has been considered as an emerging research field in machine learning and has gained more attention in recent years.

Deep Neural Networks (DNNs) are based on feed-forward structures comprised of one or more underlying hidden layers between inputs and outputs. The feed-forward architectures such as Deep Neural Networks (DNNs) and Convolutional Neural Networks (CNNs) provides efficient results for image and video processing. On the other hand, recurrent architectures such as Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) are much effective in speech-based classification such as natural language processing (NLP) and SER .

1. **LITERATURE SURVEY**
2. D.T. Manae and U.V Kulkarni proposed a SVM classifier which uses the method called Ranking SV . this has the features of periodic and spectral. And has a recognition rate of 44.4%.
3. Redounae Lhiadi proposed SVM & ANN type of classified SER. This uses a method called Multi level SVM classifier & ANN to reduce dimensionality. This has a recognition rate of 86.5%.
4. Anandhavalli muniaswami used HMM classifier which has a features of Log frequency power coefficients (LFPC), MFCC. This has a method of Log frequency power coefficients (LFPC), MFCC. This gives an average accuracy rate of 78% and best accuracy rate of 96%.
5. Siddique latif has used Decision trees and random forest classifiers. this consists of Linguistic , spectral-related , contour related, tone-based and /or vowel-related features. It’s method is to ensemble random forest to trees (ERFT rees) method with a high number of features. This gives a Best 82.54% & worst 16% accuracy rates.
6. M. Shamim Hossain used Bayesian Logistic Regression, SVM classifier using Hierarchical structure for binary decision tree methods . this has a feature of large-margin. This gives a 70.1% & 65.1% for two and five class as an accuracy.
7. Nur Farhan Hordri used SVM& RBF classifiers that has features like Modulation spectral features (MSFs). Uses the methods like Modulation filter bank & auditory filter bank for speech decomposition, SVM & RBF for classification. This gave an accuracy of 91.6%.

**3. PROPOSED WORK**

After many considerations of previous works of SER’s, we find that it is easy to generate the emotion using Deep learning techniques. There are many ways in Deep learning to extract the emotions from speech. The deep learning uses neural networks to process the given data. This helps us to find the results a way faster than others.

In our application we are using a technique of Deep learning called LSTM (Long-Short Term Memory).

* 1. **LSTM DEEP LEARNING**

Long-Short Term Memory(LSTM) is an artificial recurrent neural network(RNN) which is an architecture used in the field of deep learning. LSTM neural networks are well suited to classify, process and make predictions based on time- series data, since there can be lags of unknown duration between important events in time series. LSTM was designed by Hochreiter & Schmidhuber. It tackled the problem of long-term dependencies of RNN in which the RNN cannot predict the word stored in the long-term memory but can give more accurate predictions from the recent information.

Some of the famous applications of LSTM includes:

Language Modelling

Machine Translation

Image Captioning

Handwriting generation

Question Answering Chatbots

In concept, an LSTM recurrent unit tries to “remember” all the past knowledge that the network is seen so far and to “forget” irrelevant data. This is done by introducing different activation function layers called “gates” for different purposes. Each LSTM recurrent unit also maintains a vector called the Internal Cell State which conceptually describes the information that was chosen to be retained by the previous LSTM recurrent unit.

**4 .COMPONENTS AND SOFTWARE REQUIREMENTS**

**REQUIREMENTS:**

|  |  |
| --- | --- |
| **HARDWARE REQUIREMENTS** | **SOFTWARE REQIREMENTS** |
| * Laptop/Desktop with Processor 2.8 GHz and above * RAM 512MB and above | * Kaggle’s python environment * Packages and data sets related to SER . * VS code |

# REQUIREMENTS

Kaggle is a platform that provides interfaces for any type of languages like python ,java, c, c+ + etc.., since our project uses python we need python environment in order to run our code. So after logging into the Kaggle’s website we need to select python environment and then we can start coding.

VS code(virtual studio code) it is an editor which is used to edit our code and can find the syntaxes and errors in a simple way. After editing we can paste the code in the Kaggle’s environment.

For SER we specially need some of the data sets and packages . the TESS data sets consists of 5600 files of emotions. There are files of emotions like fear, happy, sad, disgust, ps, anger etc.., consists of 7 emotions of each 800 files.

**List of packages that are needed for SER:**

1.pandas

2.numpy

3.os

4.seaborn

5.matplotlib.pyplot

6.librosa

7.librosa.display

8.IPython.display **import** Audio

9.warnings

**List of data sets:**

**1.TESS Toronto emotional speech set data:**

OAF\_Fear

OAF\_Pleasant\_surprise

OAF\_Sad

OAF\_angry

OAF\_disgust

OAF\_happy

OAF\_neutral

TESS Toronto emotional speech set data

YAF\_angry

YAF\_disgust

YAF\_fear

YAF\_happy

YAF\_neutral

YAF\_pleasant\_surprised

YAF\_sad

**2. RAVDESS data set**

**5.SOURCE CODE:**

**import** numpy **as** np *# linear algebra*

**import** pandas **as** pd *# data processing, CSV file I/O (e.g. pd.read\_csv)*

#importing modules

**import** pandas **as** pd

**import** numpy **as** np

**import** os

**import** seaborn **as** sns

**import** matplotlib.pyplot **as** plt

**import** librosa

**import** librosa.display

**from** IPython.display **import** Audio

**import** warnings

warnings**.**filterwarnings('ignore')

#**loading the data**

paths **=** []

labels **=** []

**for** dirname, \_, filenames **in** os**.**walk('/kaggle/input'):

**for** filename **in** filenames:

paths**.**append(os**.**path**.**join(dirname, filename))

label **=** filename**.**split('\_')[**-**1]

label **=** label**.**split('.')[0]

labels**.**append(label**.**lower())

print('Dataset is Loaded')

#selecting number of paths

paths[:5]

#showing the lables

labels[:5]

*## Create a dataframe*

df **=** pd**.**DataFrame()

df['speech'] **=** paths

df['label'] **=** labels

df**.**head()

df['label']**.**value\_counts()

#**EXPLORATORY DATA ANALYSIS**

sns**.**countplot(df['label'])

**def** waveplot(data, sr, emotion):

plt**.**figure(figsize**=**(10,4))

plt**.**title(emotion, size**=**20)

librosa**.**display**.**waveplot(data, sr**=**sr)

plt**.**show()

**def** spectogram(data, sr, emotion):

x **=** librosa**.**stft(data)

xdb **=** librosa**.**amplitude\_to\_db(abs(x))

plt**.**figure(figsize**=**(11,4))

plt**.**title(emotion, size**=**20)

librosa**.**display**.**specshow(xdb, sr**=**sr, x\_axis**=**'time', y\_axis**=**'hz')

plt**.**colorbar()

#**showing spectrogram of all the emotions**

emotion **=** 'fear'

path **=** np**.**array(df['speech'][df['label']**==**emotion])[0]

data, sampling\_rate **=** librosa**.**load(path)

waveplot(data, sampling\_rate, emotion)

spectogram(data, sampling\_rate, emotion)

Audio(path)

emotion **=** 'angry'

path **=** np**.**array(df['speech'][df['label']**==**emotion])[1]

data, sampling\_rate **=** librosa**.**load(path)

waveplot(data, sampling\_rate, emotion)

spectogram(data, sampling\_rate, emotion)

Audio(path)

emotion **=** 'disgust'

path **=** np**.**array(df['speech'][df['label']**==**emotion])[0]

data, sampling\_rate **=** librosa**.**load(path)

waveplot(data, sampling\_rate, emotion)

spectogram(data, sampling\_rate, emotion)

Audio(path)

emotion **=** 'neutral'

path **=** np**.**array(df['speech'][df['label']**==**emotion])[0]

data, sampling\_rate **=** librosa**.**load(path)

waveplot(data, sampling\_rate, emotion)

spectogram(data, sampling\_rate, emotion)

Audio(path)

emotion **=** 'sad'

path **=** np**.**array(df['speech'][df['label']**==**emotion])[0]

data, sampling\_rate **=** librosa**.**load(path)

waveplot(data, sampling\_rate, emotion)

spectogram(data, sampling\_rate, emotion)

Audio(path)

emotion **=** 'ps'

path **=** np**.**array(df['speech'][df['label']**==**emotion])[0]

data, sampling\_rate **=** librosa**.**load(path)

waveplot(data, sampling\_rate, emotion)

spectogram(data, sampling\_rate, emotion)

Audio(path)

emotion **=** 'happy'

path **=** np**.**array(df['speech'][df['label']**==**emotion])[0]

data, sampling\_rate **=** librosa**.**load(path)

waveplot(data, sampling\_rate, emotion)

spectogram(data, sampling\_rate, emotion)

Audio(path)

**# Feature Extraction**

**def** extract\_mfcc(filename):

y, sr **=** librosa**.**load(filename, duration**=**3, offset**=**0.5)

mfcc **=** np**.**mean(librosa**.**feature**.**mfcc(y**=**y, sr**=**sr, n\_mfcc**=**40)**.**T, axis**=**0)

**return** mfcc

extract\_mfcc(df['speech'][0])

X\_mfcc **=** df['speech']**.**apply(**lambda** x: extract\_mfcc(x))

X\_mfcc

X **=** [x **for** x **in** X\_mfcc]

X **=** np**.**array(X)

X**.**shape

*## input split*

X **=** np**.**expand\_dims(X, **-**1)

X**.**shape

**from** sklearn.preprocessing **import** OneHotEncoder

enc **=** OneHotEncoder()

y **=** enc**.**fit\_transform(df[['label']])

y **=** y**.**toarray()

y**.**shape

## # Create the LSTM Model

**from** keras.models **import** Sequential

**from** keras.layers **import** Dense, LSTM, Dropout

model **=** Sequential([

LSTM(123, return\_sequences**=False**, input\_shape**=**(40,1)),

Dense(64, activation**=**'relu'),

Dropout(0.2),

Dense(32, activation**=**'relu'),

Dropout(0.2),

Dense(7, activation**=**'softmax')

])

model**.**compile(loss**=**'categorical\_crossentropy', optimizer**=**'adam', metrics**=**['accuracy'])

model**.**summary()

***# Train the model***

history **=** model**.**fit(X, y, validation\_split**=**0.2, epochs**=**100, batch\_size**=**512, shuffle**=True**)

## #Ploting the results

epochs **=** list(range(100))

acc **=** history**.**history['accuracy']

val\_acc **=** history**.**history['val\_accuracy']

plt**.**plot(epochs, acc, label**=**'train accuracy')

plt**.**plot(epochs, val\_acc, label**=**'val accuracy')

plt**.**xlabel('epochs')

plt**.**ylabel('accuracy')

plt**.**legend()

plt**.**show()

loss **=** history**.**history['loss']

val\_loss **=** history**.**history['val\_loss']

plt**.**plot(epochs, loss, label**=**'train loss')

plt**.**plot(epochs, val\_loss, label**=**'val loss')

plt**.**xlabel('epochs')

plt**.**ylabel('loss')

plt**.**legend()

plt**.**show()

**6.NOVELITY OF THE PROJECT**

We have seen different speech emotion recognition projects in python , but what makes a difference in our project is we have used DEEP LEARNING technology to build this application. In this new generation , Deep learning is widely used in every aspect , so that we made a deep learning model for speech emotion recognition which can extract the frequencies of the given audio files , then analyze the emotion and finally recognize the emotion.

**7.EXPECTED RESULT**

These are some results which we can expect in this application:

Fig 7.1Loading the data sets and showing the paths of files

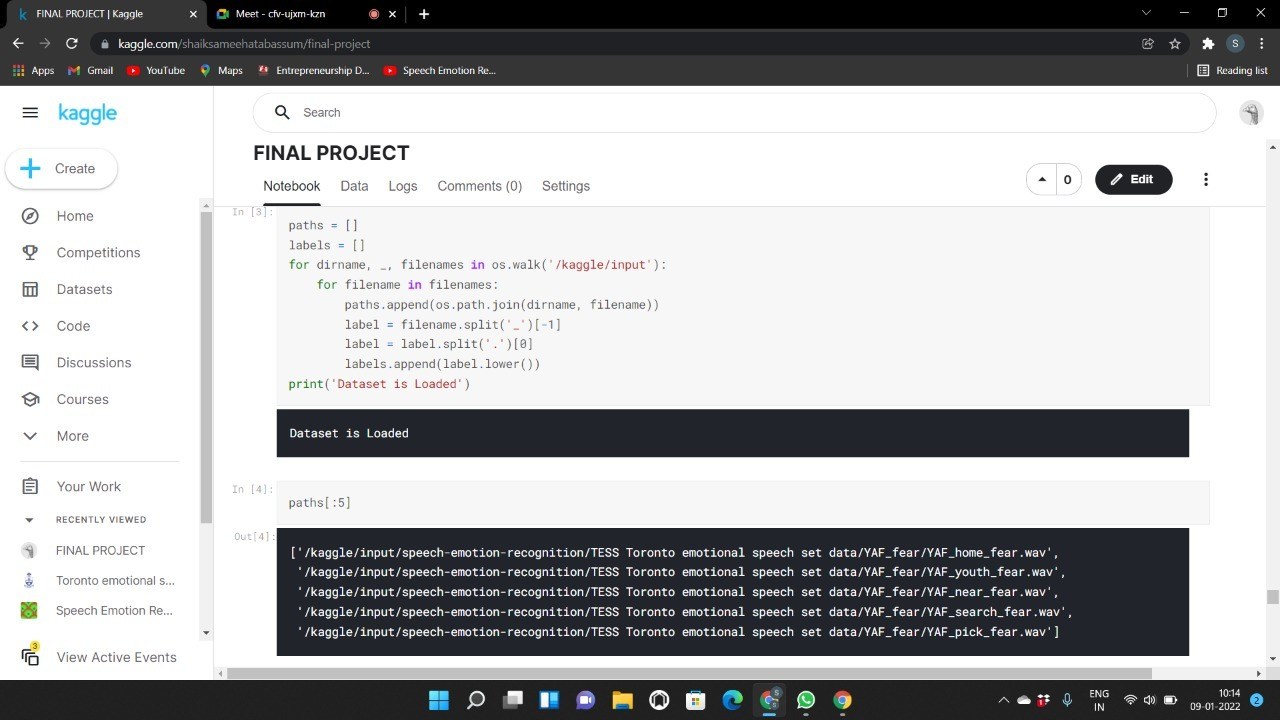
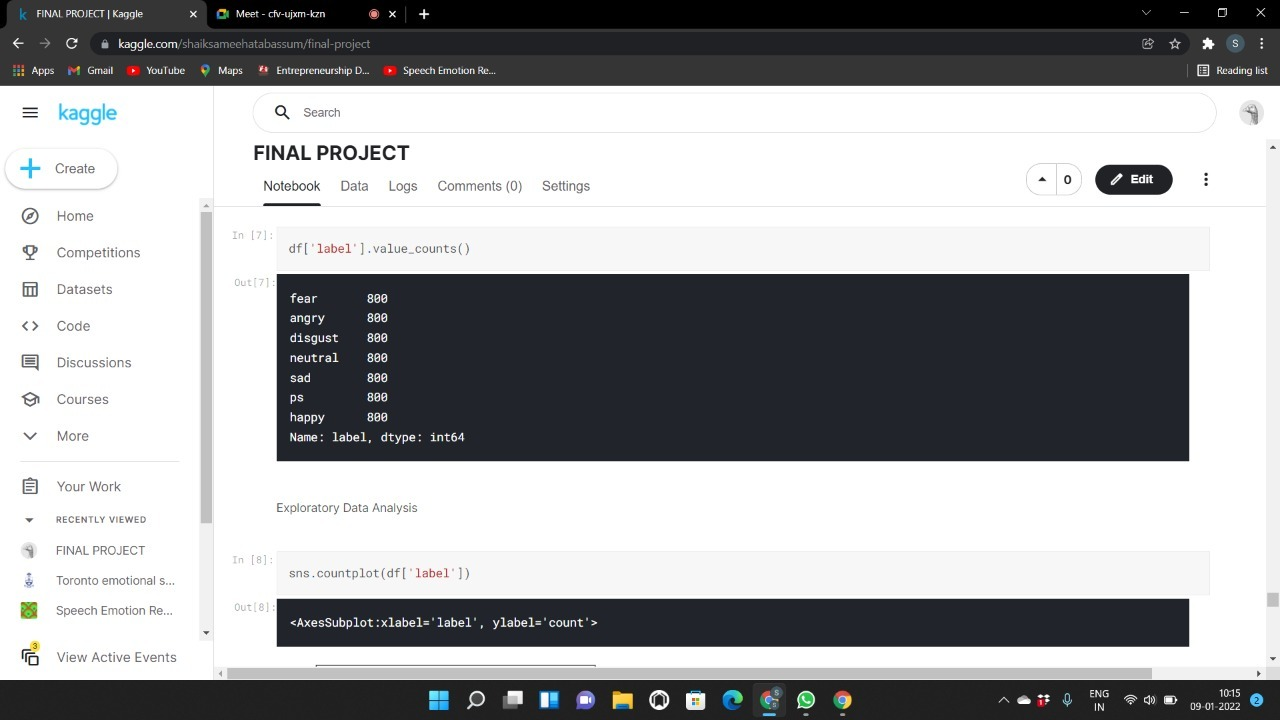


Fig 7.2. Analysis of the emotions:



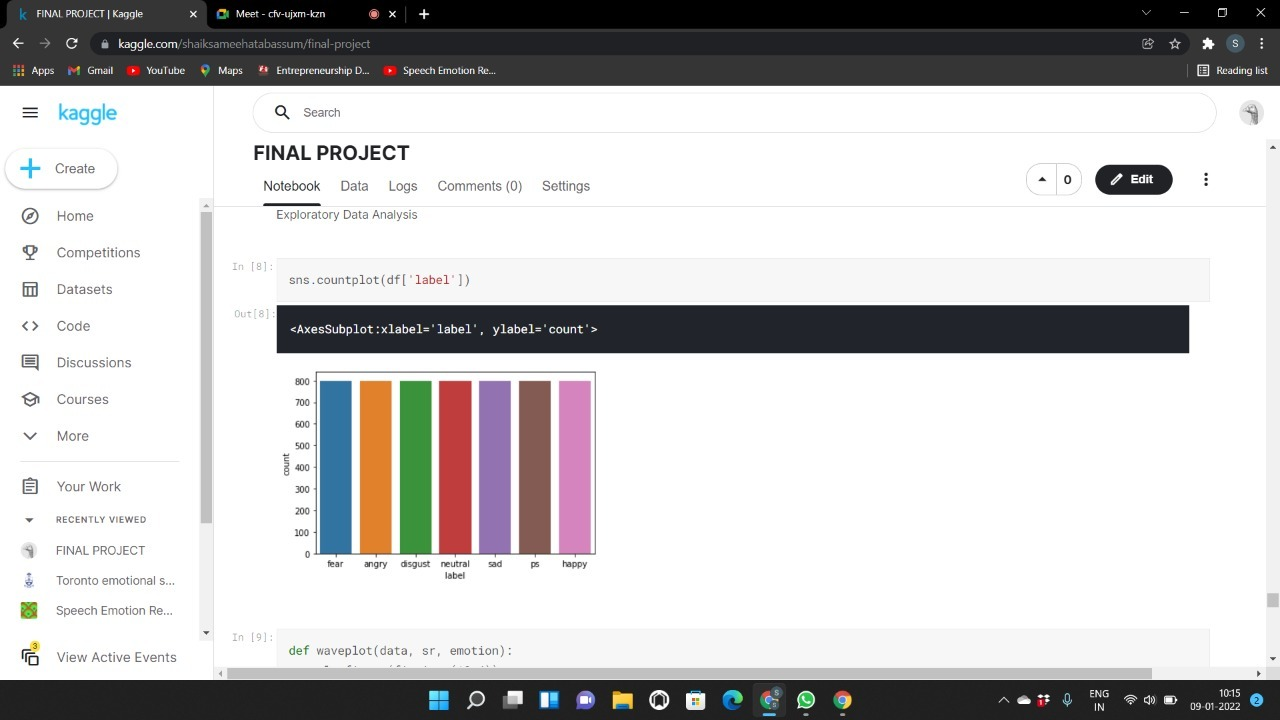


Fig 7.3 Plotting of the number of files and It’s emotionson Y and X axis respectively.

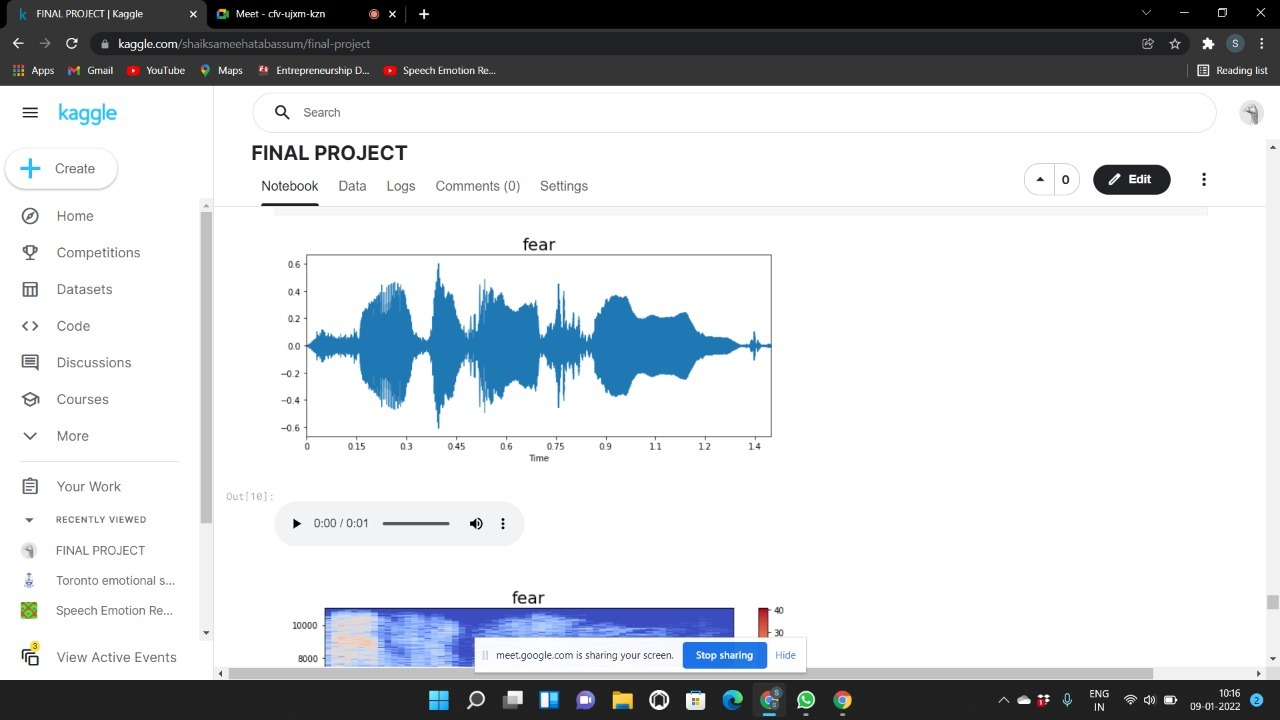
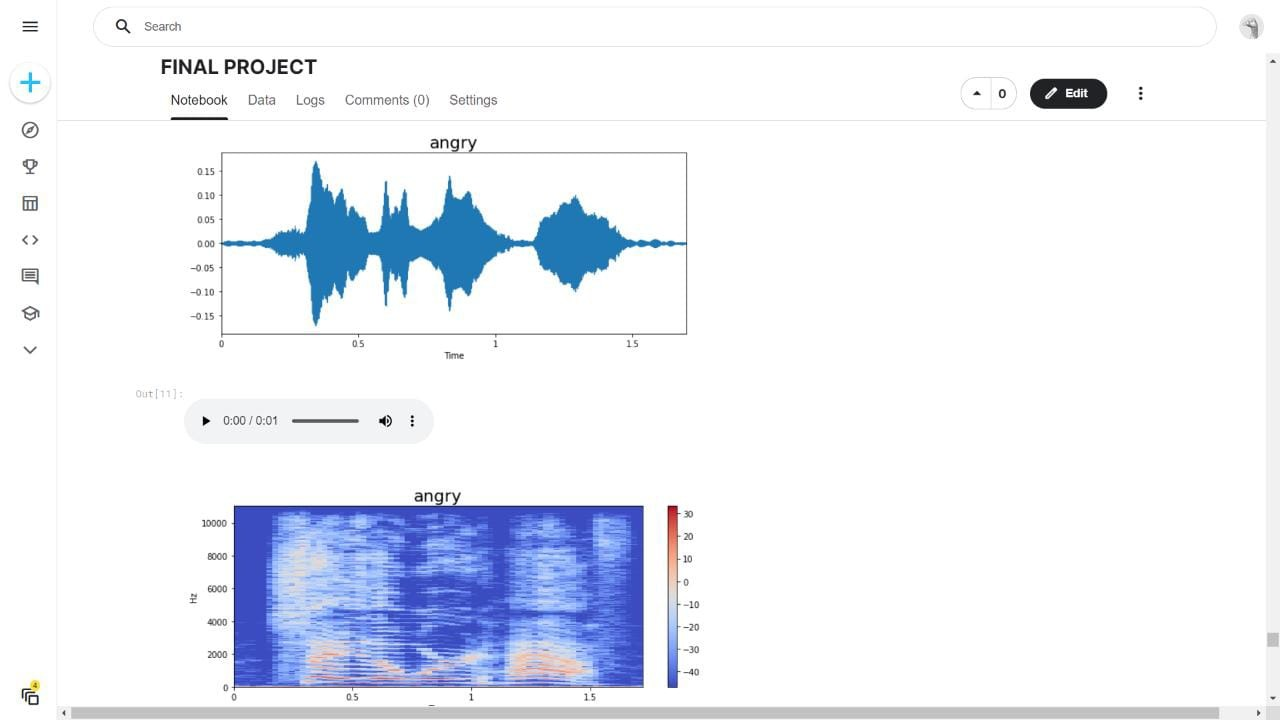


Fig 7.4 Graph when fear audio is given

Fig 7.5Spectrogram representation of result



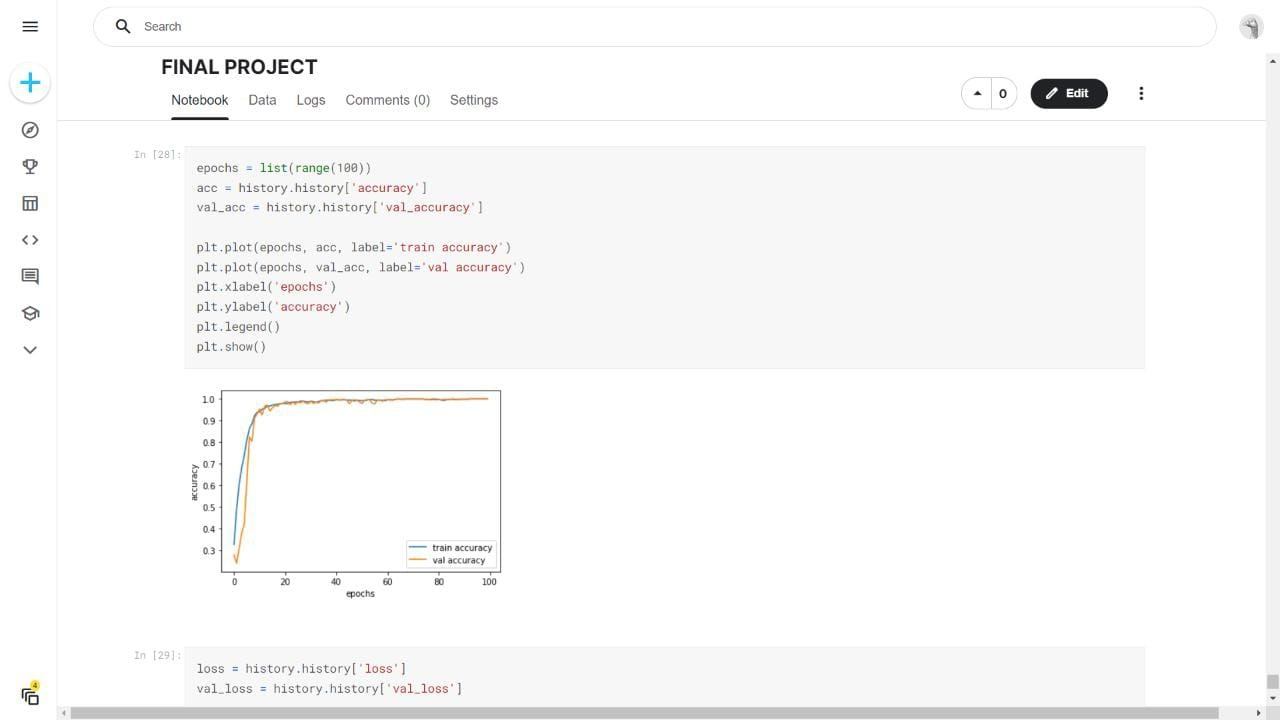


Fig 7.6 representation of loss and accuracies values

**CONCLUSION AND JUSTIFICATION**

The recent interest in speech emotion recognition research has seen applications in call center analytics, human machine and human robot interfaces, multimedia retrieval, surveillance tasks, behavioural health informatics, and improved speech recognition.

The overview of SER methods are discussed for extracting audio features from speech sample, various classifier algorithms are explained briefly.

This code shows the LSTM classification of provided audio files and also represents the results of data given in spectrograms. This is very much useful to determine the frequencies of audio signals over time. With this we can easily determine the emotion of the audio of person.

Speech Emotion Recognition has a promising future and its accuracy depends upon the emotional speech database , combination of features extracted from those database for training the model, types of classification algorithm used to classify the emotions in appropriate emotion class (e.g. happy, sad, anger, surprise etc.).

This study aims to provide a simple guide to the beginner who’s carried out their research in the speech emotion recognition.

**REFERENCES:**

S. G. Koolagudi and K. S. Rao, “Exploring speech features for classifying emotions along valence dimension,” in The 3rd international Conference on Pattern Recognition and Machine Intelligence (PReMI- 09), Springer LNCS (S. C. et al., ed.), (IIT Delhi), pp. 537–542, Springer-verlag, Heidelberg, Germany, December 2009.

S. G. Koolagudi, S. ray, and K. S. Rao, “Emotion classification based on speaking rate,” in The 3rd International Conference on Contemporary Computing, (Noida, India), JIIT university and University of Florida, August 2010.

L. R. Rabiner and B. H. Juang, Fundamentals of Speech Recognition. Englewood Cliffs, New Jersy: Prentice-Hall, 1993.

D. Ververidis and C. Kotropoulos, “Emotional speech recognition: Resources, features, and methods,” Speech Communication, vol. 48, p. 11621181, 2006.

S. R. M. Kodukula, Significance of Eecitation Source Information for Speech Analysis. PhD

Ashish B. Ingale, Dr.D.S.Chaudhari, “SPEECH EMOTION RECOGNITION USING HIDDEN MARKOV MODEL AND SUPPORT VECTOR MACHINE”.

Muzaffar Khan, Tirupati Goskula, Mohmmed Nasiruddin ,Ruhina Quazi, “Comparison between k-nn and svm method for speech emotion recognition”, International Journal on Computer Science and Engineering (IJCSE).

Anurag Kumar, Parul Agarwal, Pranay Dighe1, “ Speech Emotion Recognition by AdaBoost Algorithm and Feature Selection for Support Vector Machi